

ProjectDevelopmentPhase

DeliveryofSprint- 4

Date	18 November 2022
TeamID	PNT2022TMID32337
ProjectName	AI-based discourse for Banking Industry

Creating Assistant & Integrate With Flask WebPage

You will be creating a banking bot in this activity that has the following capabilities

1. The Bot should be able to guide a customer to create a bank account.
2. The Bot should be able to answer loan queries.
3. The Bot should be able to answer general banking queries.
4. The Bot should be able to answer queries regarding netbanking.
5. With the help of this bot ,you can get all the required details related to banking.

Let us build our flask application which will be running in our local browser with a user interface.

In the flask application ,users will interact with the chatbot, and based on the user queries they willgettheoutcomes.

Build PythonCode

1:ImportingLibraries

The first step is usually importing the libraries that will be needed in the program.

```
from flask import Flask, render_template
```

Importing the flask module into the project is mandatory. An object of the Flask class is our WSGI application.Flask constructor takes the name of the current module(name).

2 . Creating our flask application and loading

```
app = Flask(__name__)
```

3. Routing to the HtmlPage

Here, the declared construct or is used to route to the HTML page created earlier.

The `` route is bound with the bot function. Hence, when the home page of a webserver is opened in the browser, the HTML page will be rendered.

```
@app.route('/')  
def bot():  
    return render_template('chatbot.html')
```

Main Function

This is used to run the application in localhost.

```
if __name__ == '__main__':  
    app.run()
```

Build HTMLCode

- We use HTML to create the front-end part of the webpage.
- Here, we have created chatbot.html
- Chatbot.html displays the home page which integrates with Watson Assistant.
- A simple HTML page is created. Auto-generated source code from IBM Watson Assistants is copied and pasted inside the body tag

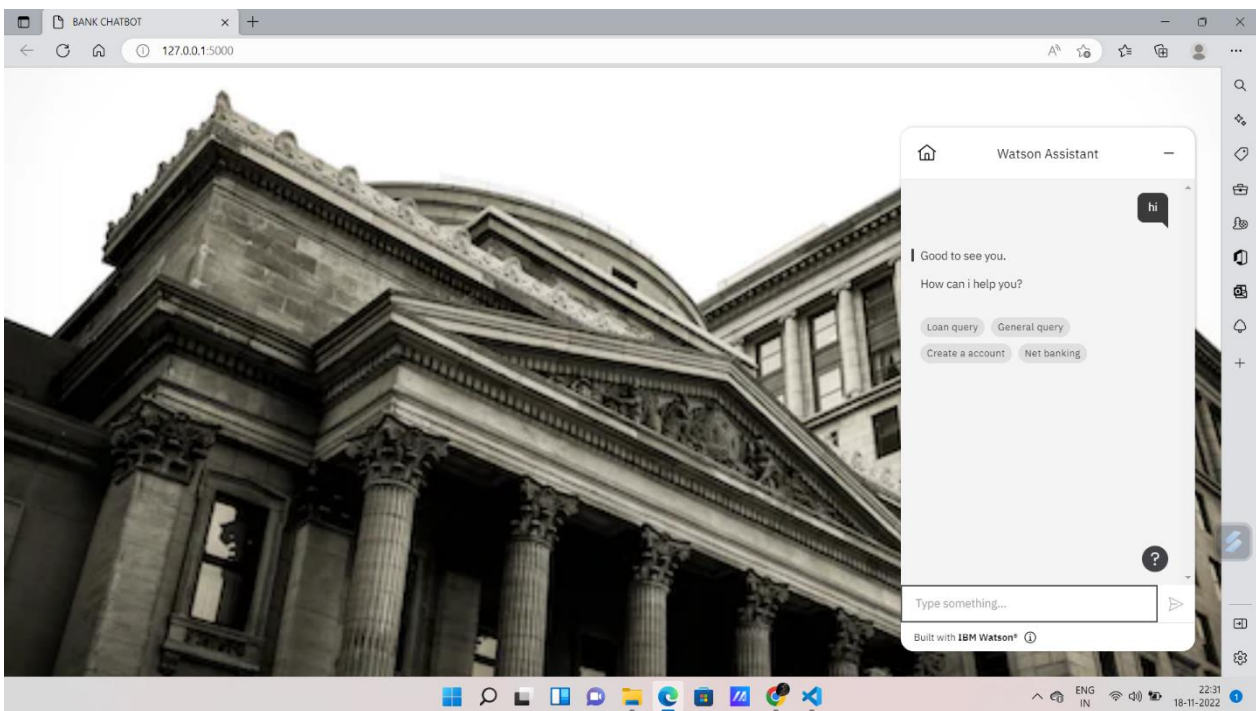
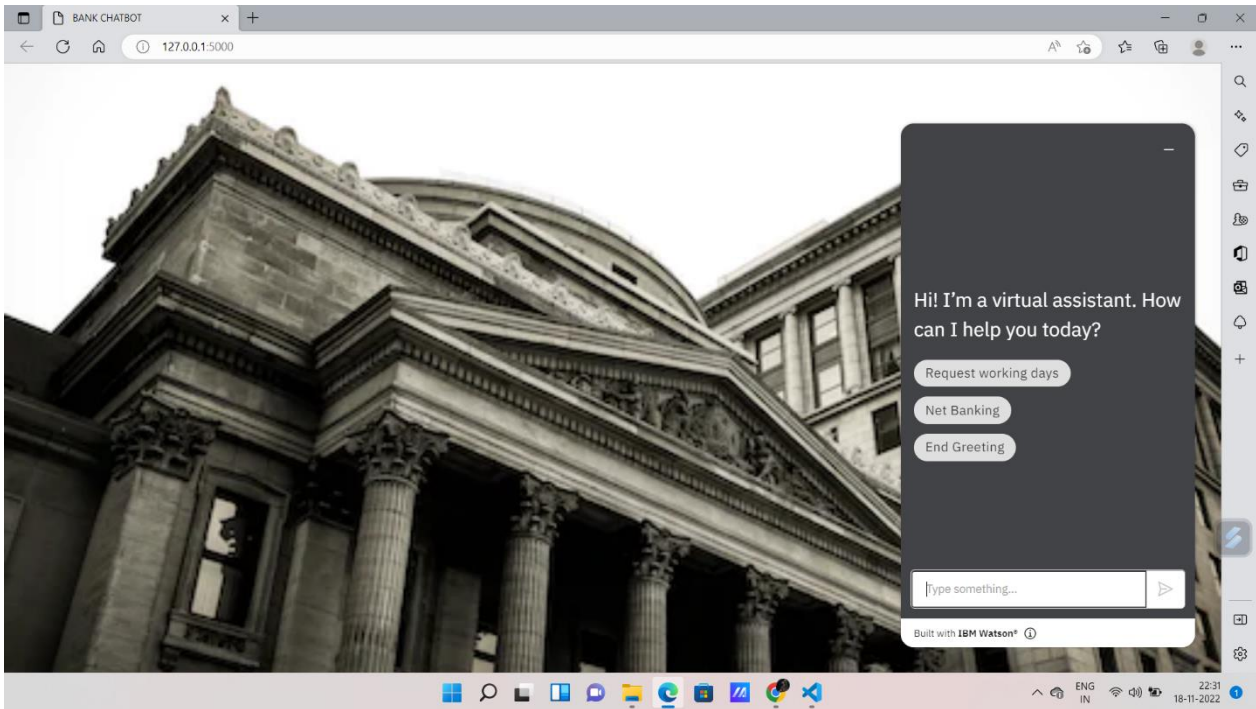
Run The Application

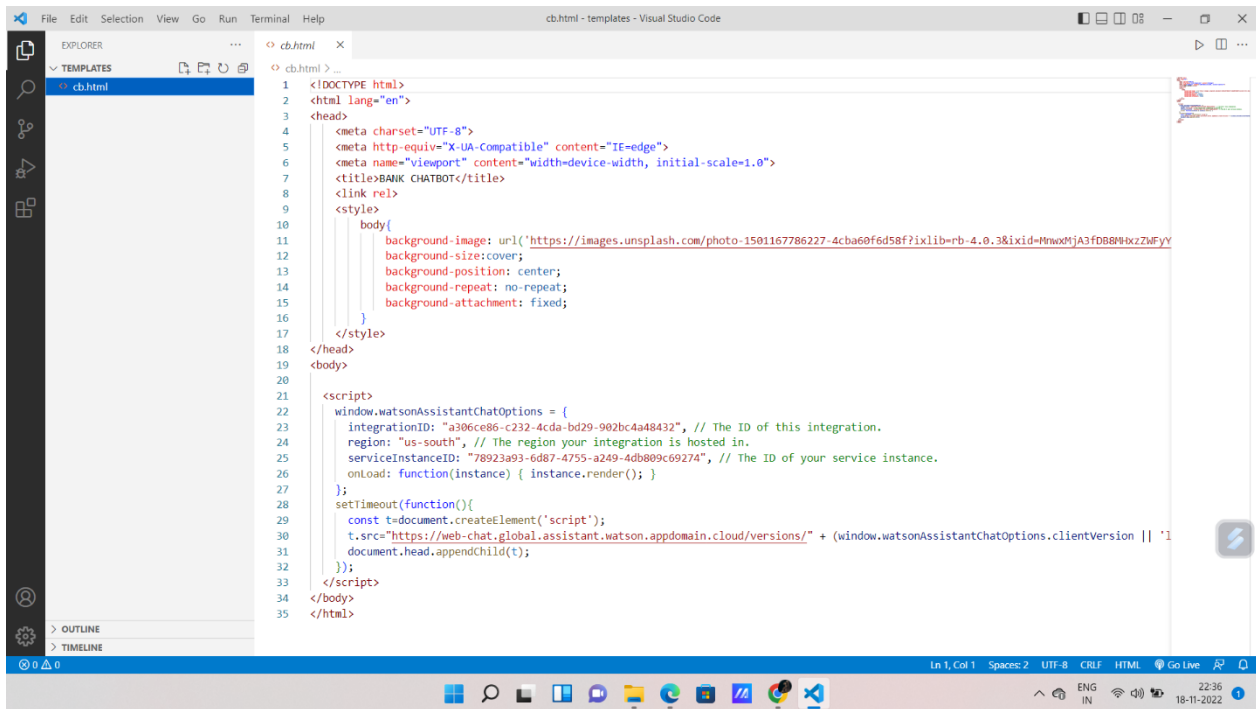
Run the application

- Open the command prompt from the start menu.
- Navigate to the folder where your chatbot.
- Now type the "python Run.py" command.

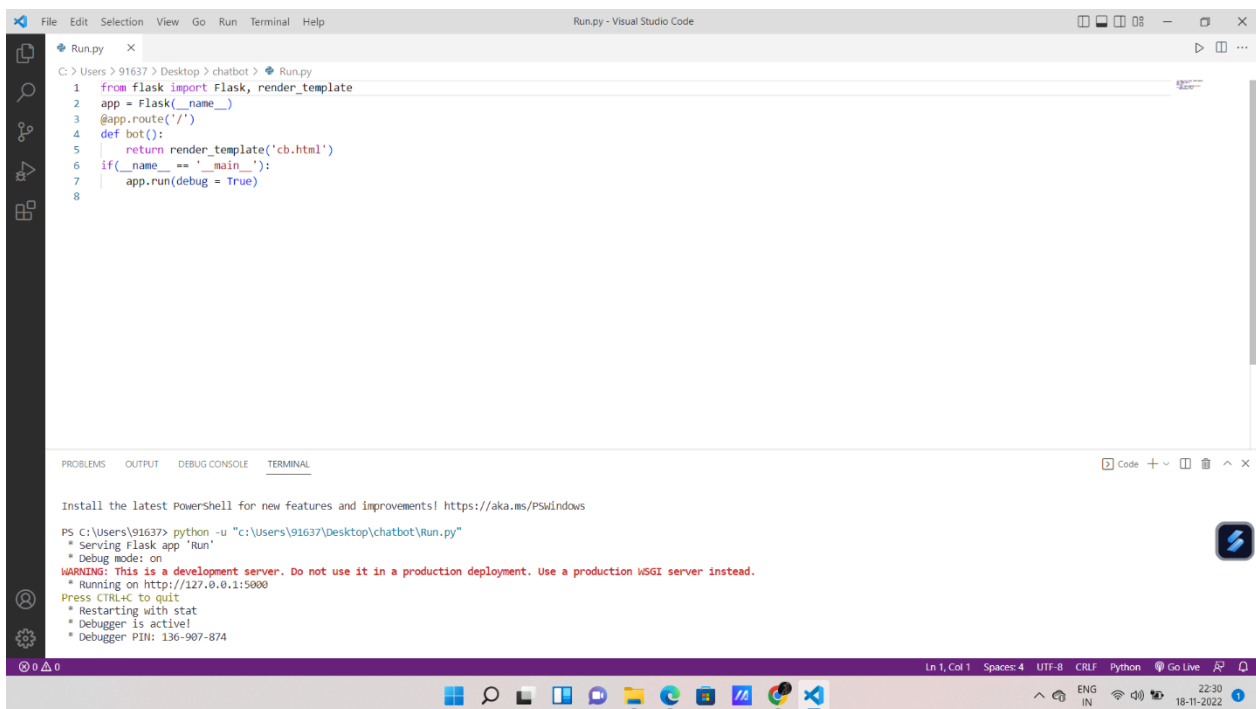
- It will show the localhost where your app is running on `http://127.0.0.1:5000/`
- Copy that localhost URL and open that URL in the browser. It does navigate me to where you can view your webpage.

Then it will run on localhost:5000





```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>BANK CHATBOT</title>
8   <link rel>
9   <style>
10     body{
11       background-image: url('https://images.unsplash.com/photo-1501167786227-4cba60fd58f7?ixlib=rb-4.0.3&id=Mnw0xjA3fDB8M9kzZWYy');
12       background-size: cover;
13       background-position: center;
14       background-repeat: no-repeat;
15       background-attachment: fixed;
16     }
17   </style>
18 </head>
19 <body>
20
21   <script>
22     window.watsonAssistantChatOptions = {
23       integrationID: "a306ce86-c232-4cda-bd29-902bc4a48432", // The ID of this integration.
24       region: "us-south", // The region your integration is hosted in.
25       serviceInstanceID: "78923a93-6d87-4755-a249-4db09c69274", // The ID of your service instance.
26       onLoad: function(instance) { instance.render(); }
27     };
28     setTimeout(function(){
29       const t=document.createElement('script');
30       t.src="https://web-chat.global.assistant.watson.appdomain.cloud/versions/" + (window.watsonAssistantChatOptions.clientVersion || '1.0.0') + "/web-chat.js?container=chatbot";
31       document.head.appendChild(t);
32     });
33   </script>
34 </body>
35 </html>
```



```
1 from flask import Flask, render_template
2 app = Flask(__name__)
3 @app.route('/')
4 def bot():
5     return render_template('cb.html')
6 if __name__ == '__main__':
7     app.run(debug = True)
8
```

Install the latest PowerShell for new features and improvements! <https://aka.ms/PSWindows>

PS C:\Users\91637> python -u "C:\Users\91637\Desktop\chatbot\Run.py"

* Serving Flask app 'Run'

* Debug mode: on

WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.

* Running on <http://127.0.0.1:5000>

Press CTRL+C to quit

* Restarting with stat

* Debugger is active!

* Debugger PIN: 136-907-874